

работы. Они, как правило, не имеют степени, а, следовательно, могут рассчитывать только на самую низкую ставку оплаты труда. Введя такую «степенную функцию» начисления зарплаты, мы обрекаем себя на замкнутый мир внутри своей вузовской системы без участия специалистов промышленности.

Вновь вводимая система оплаты труда еще более усугубит проблему, т.к. на стимулирующую надбавку, а именно эта надбавка зависела от интенсивности, качества и результатов труда отводится не более 30 процентов от фонда заработной платы.

Все вышесказанное приводит к неутешительным выводам, что процессы высшей школы далеки от оптимальности и, если они не будут «фундаментально, существенно и радикально» перестроены (т.е. проведен их реинжиниринг), то не помогут никакие другие инновационные (информационные) технологии, которые мы пытаемся внедрять, потому что фундамент, на котором все строится, очень шаткий и создан с ошибками.

-
1. Хаммер М., Чампи Дж. Реинжиниринг корпорации: Манифест революции в бизнесе. /Пер. с англ. - СПб.: Издательство С.-Петербургского университета, 1997. - 332 с.

Игнатова Ю.А.

МОДЕЛЬ ДАННЫХ РЕПОЗИТОРИЯ ПОНЯТИЙ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СРЕДЫ ВУЗА

Yuliya.Ignatova@vvsu.ru

*Владивостокский государственный университет экономики и сервиса
г. Владивосток*

Статья посвящена методам организации работы с корпоративными данными, программной реализацией одного из которых является репозиторий понятий информационного пространства вуза. В настоящей работе рассматривается модель данных репозитория понятий КИС ВГУЭС, в частности описывается структура и способы поддержания целостности данных, а так же основные принципы их извлечения и манипулирования в рамках этой модели.

The concepts repository of university corporative information environment as a data management method is the main subject of the article. The concepts repository data model which describes the aspects of data structure, data integrity and data processing, is considered in this article.

Корпоративная информационная среда (КИС) Владивостокского государственного университета экономики и сервиса (ВГУЭС) помимо совокупности телекоммуникационного, сетевого оборудования и программного обеспечения, включает так же и данные, которые обрабатываются приложениями КИС и с которыми взаимодействуют пользователи КИС. В процессе организации работы с такими данными часто

требуется оперировать некоторыми понятиями информационной среды, не создавая при этом отдельных программных средств управления этими понятиями. Для поддержания рассматриваемого процесса необходимо удобное и простое в использовании единое инструментальное программное средство, позволяющее эффективно организовать создание, хранение, представление и управление понятиями информационной среды. Для таких целей в КИС ВГУЭС используется репозиторий понятий. Подсистема хранения данных репозитория содержит полное описание понятий КИС, а также экземпляры тех понятий, которые являются внутренними по отношению к локальному источнику данных репозитория.

Одним из требований к КИС вуза является возможность ее интеграции со сторонними информационными системами, что, в свою очередь, определяет необходимость интеграции данных сторонних систем в репозиторий понятий. Для таких целей репозиторий поддерживает работу с внешними понятиями по отношению к локальному источнику данных репозитория. Внешние понятия, в отличие от внутренних – хранимых непосредственно в репозитории, имеют в нем только описание, а экземпляры хранятся в источниках данных сторонних информационных систем. Для упрощения работы с экземплярами внешних понятий, а так же для обеспечения большей наглядности и оперативности, реализован алгоритм импортирования внешних данных (данных сторонних информационных систем) непосредственно в репозиторий, с последующим хранением и поддержанием в актуальном состоянии. Таким образом, появляется возможность работать с данными внешних систем, организованных в качестве внутренних данных репозитория.

При описании атрибутов понятия формируются различные ограничения на атрибуты: на тип данных, на допустимый диапазон, на допустимый домен, ограничения в виде регулярного выражения (в том числе логического выражения с результатом истина или ложь). При описании отношений в репозитории понятий разрешено определять характеристики отношений: симметричности, транзитивности, инверсности. В репозитории, так же, формируются отношения доступа на чтение/запись между внешними понятиями и проектами. Отношения внешнего понятия с проектом определяется тем, в каком проекте предполагается редактировать экземпляры понятия, в связи с тем, что в каждом проекте могут быть и внутренние и внешние понятия.

Поскольку репозиторий должен обеспечивать работу с динамически увеличивающимся количеством экземпляров понятий информационного пространства, имеющих при этом переменную структуру (различный состав атрибутов различных типов), подсистема хранения данных репозитория должна отвечать не столько требованиям оптимальности, сколько универсальности. Так, использование классической методики проектирования баз данных [1], – предполагающей создание отдельной таблицы в базе для каждой описываемой сущности и далее, в процессе нормализации, выделение индивидуальных таблиц для хранения атрибутов

сущностей, – потребовало бы постоянной реорганизации базы данных репозитория и, со временем, привело бы к неограниченному разрастанию количества ее объектов. В связи с этим, в качестве схемы проектирования базы данных репозитория был выбран один из вариантов реализации модели “Сущность-атрибут-значение” (EAV, «Entity-attribute-value» data model [2]) на основе реляционной СУБД (среда MS SQL Server 2000). Модель EAV представляет собой некую универсальную структуру описания данных, которая в самом общем смысле подразумевает хранение всех данных в виде так называемых таблиц-справочников: справочника сущностей, справочника атрибутов и справочника значений. Для различных типов данных (строковых, числовых и т.д.) выделяются отдельные таблицы-справочники значений, в простейшем случае используется одна таблица с применением наиболее общего формата данных. Справочник значений, помимо хранения всех данных, обеспечивает взаимосвязь с остальными таблицами-справочниками, поскольку содержит поля метаданных для ссылки на справочники сущностей и атрибутов, при этом каждый из атрибутов определенной сущности отображается на одну запись в справочнике значений. Таким образом, весь массив хранимых данных, образованный совокупностью значений таблиц-справочников, организуется фактически в один линейный список, что обеспечивает принципиальное единообразие хранения всех данных, а, следовательно, и существенную гибкость (настраиваемость) системы, разрабатываемой на основе данной модели.

В рамках рассматриваемого подхода, база данных репозитория понятий КИС ВГУЭС включает совокупность следующих реляционных таблиц (рис.1):

- таблица D_DictList – используется в качестве справочника сущностей, ключевое поле IdObjectDepend однозначно идентифицирует каждый из экземпляров понятий информационного пространства. Поле IdDictType – определяет принадлежность экземпляра к классу внешних или внутренних понятий по отношению к локальному источнику данных репозитория; поле IdObjectGroup – ссылка на справочник групп экземпляров понятий;
- таблица D_Dictionary – используется в качестве справочника групп экземпляров понятий информационного пространства, созданных с целью объединения экземпляров по тематическому признаку; ключевое поле Ids обеспечивает взаимосвязь между понятиями на уровне их метаописания (тип связи понятие-родитель → понятие-потомок);
- таблица D_Params – используется в качестве справочника атрибутов, ключевое поле IdParam уникально идентифицирует каждый из атрибутов экземпляров понятий, поле IdType – определяет тип данных атрибута, поле IdObject – используется для ссылки на справочник сущностей;
- таблица D_Item – используется в качестве справочника групп значений атрибутов (записей). Поскольку доминирующей задачей репозитория

является возможность использования экземпляров понятий информационного пространства для хранения условно постоянных наборов данных, дополняемых (расширяемых) в процессе эксплуатации системы, – т.е. каждому атрибуту определенного экземпляра соответствует более одного значения (множество значений) – появляется необходимость определять группы взаимосвязанных значений атрибутов в рамках отдельных записей. Для идентификации записей используется ключевое поле `ItemId`, поле `IdObject` – для ссылки на справочник сущностей;

- таблица `D_ParamDetails` – используется в качестве справочника значений для хранения всего массива данных. Поле `ParamText` непосредственно используется для хранения данных строкового типа (длиной до 8000 символов), поле `ParamMoney` – для числовых данных, поле `ParamDate` – для данных типа дата/время, поле `ParamValue` – для организации связей с другими понятиями (устанавливается взаимосвязь между понятиями на уровне данных, тип атрибут родительского понятия→атрибут понятия-потомка), поле `IdObject` – ссылка на справочник сущностей, поле `IdParam` – ссылка на справочник атрибутов, поле `InfoId` – ссылка на справочник записей;
- таблица `D_LongData` – используется для хранения текстовых блоков данных длиной до 2147483647 символов.

Логическая схема данных (рис.1), помимо описания собственно структуры, налагает на данные ряд определенных условий (ограничений), выполнение которых обеспечивает поддержание целостности данных в базе. В частности, ограничения целостности сущностей (первичные ключи и уникальные индексы), ограничения целостности атрибутов (значения по умолчанию, обязательность и необязательность значений `Null\Not Null`), а так же ограничения целостности типов данных, выполняются на декларативном уровне средствами СУБД. Правила описания понятий и организации их взаимосвязей различного типа и различного уровня (связи «понятие→понятие», «понятие→атрибут», «атрибут→атрибут» и т.д.), в т.ч. ограничения ссылочной целостности (целостность внешних ключей), принципы добавления, удаления и изменения данных, а так же дополнительные ограничения, не определенные логической схемой, выполняются на уровне прикладных программ. Логика выделяется в отдельный программный слой, реализуемый на сервере приложений, и, совместно с интерфейсом пользователя (web-интерфейсом), представляет систему управления репозиторием. Конечные наборы данных (данные самого низкого уровня, собственно пользовательские данные) содержатся в экземплярах понятий, являющихся, в свою очередь, объектами системы управления репозиторием.

Логическая схема данных репозитория понятий КИС ВГУЭС

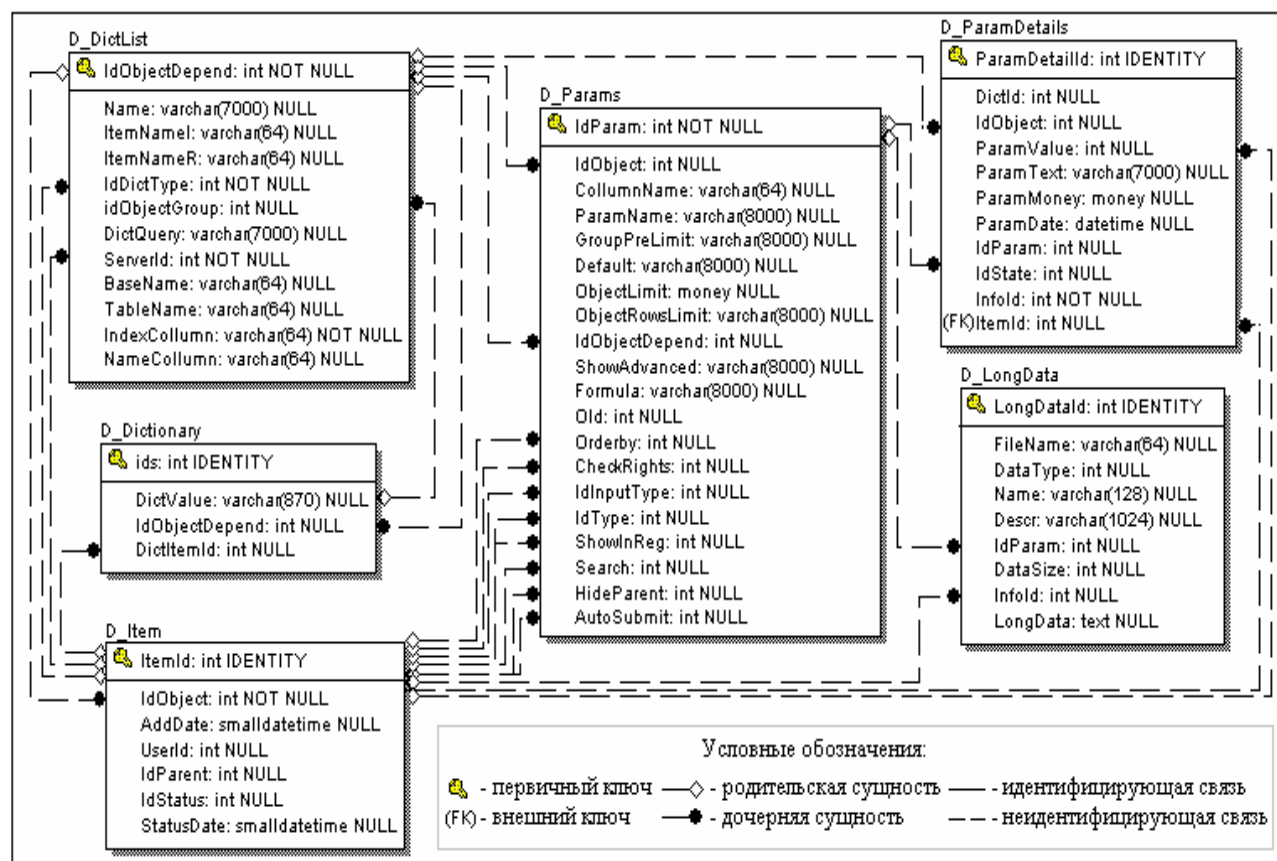


Рис. 1.

В виду однообразности модели хранения данных, служебные объекты системы управления (вплоть до справочников видов отрисовки элементов, типов полей, стилей отображения и т.п.) создаются в качестве понятий информационного пространства и хранятся в базе данных репозитория наряду с другими понятиями, таким образом, позволяя построить систему управления репозиторием на основе собственной же составной части, т.е. рекурсивно. Такой подход позволяет не создавать в базе данных репозитория отдельных таблиц метаданных для описания структуры системы управления и организации ее web-интерфейса, а так же не разрабатывать отдельных программных средств по работе с этими таблицами.

Пользователи КИС работают с данными репозитория понятий через web-интерфейс системы управления, доступный в качестве одного из сервисов корпоративного информационного портала вуза, при этом прямой доступ к таблицам базы данных для них запрещен, и структура хранения данных скрыта.

При построении интерфейса пользователя реализуется механизм по загрузке свойств и связей объектов системы из базы данных репозитория, согласно правилам описания соответствующих им экземпляров понятий, обеспечивая, таким образом, возможность работать с понятиями в привычном для пользователя «развернутом» табличном виде. При этом для каждого понятия в web-интерфейсе создается (отрисовывается) отдельная таблица и предоставляется набор элементов управления для

манипулирования понятиями и их свойствами, а так же сохранению внесенных изменений в базе данных репозитория.

Система управления репозиторием понятий использует систему управления доступом к информационным ресурсам [3] для назначения прав на редактирование, использование отдельных понятий, отдельных атрибутов понятий, а так же на редактирование структуры понятий. Так, роль «Администратор понятия» позволяет в системе управления репозиторием просматривать, редактировать экземпляры понятий. При просмотре и редактировании экземпляра, у которого в качестве свойства используется другое понятие (дочернее), система отслеживает права на использование дочернего понятия или на использование отдельных экземпляров этого понятия. Роль «Право использовать понятие», являющаяся производной от роли «Администратор понятия», позволяет задействовать определенное понятие при описании свойств других понятий, т.е. для полноценной работы со связанными понятиями. Роль «Право использовать экземпляр понятия» является производной от роли «Право использовать понятие», учитывая, что между экземпляром понятия и понятием определены отношения «являться экземпляром». Роль «Добавить экземпляр понятия» дает возможность пользователю создавать новые экземпляры данного понятия. Эта роль является базовой к роли «Право использовать экземпляр понятия». Роль «Редактировать экземпляр понятия» позволяет редактировать отдельный экземпляр понятия. Роль является производной от роли «Добавить экземпляр понятия». Такое разделение прав на пользование понятиями и их экземплярами позволяет тонко настраивать права доступа к данным в КИС.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Основы проектирования реляционных баз данных, Кириллов В.В., СПб.: ИТМО, 1997 //khpi-iip.mipk.kharkiv.edu/library/dbms/kir1
2. The EAV/CR Model of Data Representation, Nadkarni PM, J Am: Med Inform Assoc, 2000 //ycmi.med.yale.edu/nadkarni/eav_CR_contents.htm
3. Система автоматического управления доступом к информационным ресурсам вуза, Шахгельдян К.И., Крюков В.В., Гмарь Д.В., //Информационные технологии. 2006.-№2.-с.19-29